5. FUNCIONALIDAD DE VOZ: REPRODUCCIÓN, GRABACIÓN, SÍNTESIS Y RECONOCIMIENTO

5.1 INTRODUCCIÓN

En este capítulo se comentan las funciones relacionadas con el manejo de voz: reproducción, grabación, síntesis y reconocimiento. Se tratarán aspectos de implementación, como el diseño de las funciones *idle de segundo nivel* de cada una de ellas, y sus funciones 'inic' y 'fin' asociadas. También se incluyen los diagramas de flujo de las funciones *idle de segundo nivel* para facilitar el seguimiento de las explicaciones sobre su implementación.

5.2 REPRODUCCIÓN

Nos permite enviar mensajes pregrabados al usuario, los cuales deben estar guardados en ficheros como muestras de voz. La calidad de estos mensajes es mayor que la de los mensajes sintetizados, pero obliga a que los mensajes que se manden sean fijos.

DETALLES DE IMPLEMENTACION:

- **Función inic**: dependiendo del tamaño del fichero a reproducir, calcula el número de bancos del DSP que habrá que enviar (variable *n_bancos*), rellena los dos bancos del DSP e inicia la reproducción. Si el tamaño del fichero a reproducir es menor que un banco, el primer banco se rellena con ceros y el segundo banco sólo contendrá ceros.

- **Función idle**: va enviando trozos del fichero a los bancos del DSP a medida que éste los va consumiendo. El último banco a enviar se rellena con ceros. Sólo realiza la reproducción de un fichero. Es responsabilidad de la función *idle de primer nivel* llamarle tantas veces como ficheros haya que reproducir.

- **Función fin**: Cierra el fichero que se está reproduciendo y deja al DSP en el estado OCIOSO.

Cuando se alcanza el último banco, se modifica la variable del DSP que indica el final del banco que se está reproduciendo. Así evitamos reproducir el silencio final (los ceros con que se ha rellenado). Aún así es necesario rellenar con ceros, pues mientras el sistema reacciona, el DSP sigue reproduciendo.

Otra optimización se aplica a los ficheros a reproducir cuyo tamaño está comprendido entre 1 y 2 bancos. En estos casos se reparten las muestras equitativamente entre los dos bancos del DSP. La razón es evitar que se repita el primer banco en el caso de que las muestras a ocupar el segundo banco sean tan pocas que el sistema no sea capaz de detectar el cambio de banco.

5.3 GRABACIÓN

Permite guardar en un fichero de muestras de voz lo que el usuario diga durante un cierto tiempo.

DETALLES DE IMPLEMENTACION:

- **Función inic**: prepara el fichero que va a contener las muestras, guardando información en la cabecera relativa a su tamaño y frecuencia de muestreo. Calcula el número de bancos necesarios en función de la duración del fichero y de la frecuencia de muestreo indicada e inicia la grabación.

- **Función idle**: cuando detecta que el DSP ha cambiado de banco, recoge las muestras del banco que se acaba de rellenar y las guarda en el fichero que se está grabando. La grabación se finaliza, bien porque se alcanza la duración indicada, bien porque se ha detectado silencio, dependiendo del valor de uno de los parámetros de entrada que admite la instrucción asociada. En este último caso, si transcurre el intervalo de tiempo indicado y no se ha detectado silencio, también se dará por terminada la grabación.

  Si la grabación ha de finalizar por detección de silencio, se calcula la energía de cada banco que se recoge del DSP y se compara con un umbral. Para dar por finalizada la detección han de detectarse dos bancos consecutivos con una energía por debajo del umbral, siempre que antes se haya detectado un banco con energía por encima del umbral. El umbral se puede configurar modificando una entrada del fichero VISHA.INI, según se describe en el *Apendice A Manual del Usuario.*

- **Función fin**: Cierra el fichero que se está grabando y deja al DSP en estado OCIOSO. Si la grabación ha finalizado por detección de silencio, corrige la cabecera que escribió la función inic.

5.4 SÍNTESIS

Permite sintetizar mensajes que se indica en la propia instrucción como una cadena de texto, realizando la conversión texto-voz.

En este caso no se ha realizado una versión multilínea completa, debido a su complejidad y a que hay un proyecto en marcha que se encargará de ello. Como solución provisional se ha modificado el preprocesamiento, de tal forma que es posible realizarlo a través de sucesivas llamadas, llamadas que consumen muy poco tiempo. Respecto a la síntesis, también se ha modificado para que se pueda realizar a través de sucesivas llamadas, y la salida no se envía a la tarjeta VISHA, sino que se guarda en memoria. Una vez que tenemos el mensaje sintetizado en memoria, se reproduce utilizando funciones de reproducción en memoria, que son básicamente idénticas a las que se utilizan para reproducir un mensaje pregrabado a través de disco, con la salvedad de tomar las muestras de memoria.

DETALLES DE IMPLEMENTACION:

- **Función inic**: inicializa algunas variables que utilizará la función *idle*.

- **Función idle**: realiza el preprocesamiento del texto a sintetizar e inicia la reproducción de las muestras que se han guardado en memoria.

- **Función fin**: no hace nada.

El preprocesamiento es monolínea, es decir, sólo una línea puede estar haciendo llamadas a las funciones de preprocesamiento. Para sincronizar las diferentes líneas se utiliza una bandera (*flag*), de tal forma que cuando una línea desea sintetizar, lo hará si la bandera está libre, y si es así, la cerrará hasta que termine. Si no estuviera libre, se quedará esperando hasta que lo esté, lo cual ocurrirá cuando la línea que estaba en la fase de preprocesamiento termine.

Una vez realizado el preprocesamiento se debe reproducir el fichero generado en memoria con el resultado de la síntesis. Esta reproducción si es multilínea.

5.5 RECONOCIMIENTO DE VOZ

Permitirá al usuario realizar una elección entre varias posibilidades planteadas por el sistema e introducir cadenas numéricas dígito a dígito.

El reconocimiento de tonos multifrecuencia se tratará en el capítulo dedicado a la tarjeta interfaz de línea.

DETALLES DE IMPLEMENTACION:

- **Función inic**: inicializa algunas variables que utilizará la función *idle*, reserva memoria para la palabra a reconocer y su segundo candidato y pone al DSP en modo reconocimiento. Si está permitida la detección de tonos multifrecuencia se llama a su función inic.

- **Función idle**: si se ha interrumpido la reproducción o la síntesis anterior al reconocimiento, mediante el envío de un tono multifrecuencia, se intenta validar el dígito pulsado. Si está activada la detección de tonos multifrecuencia se comprueba si hay tono (cuando en el diagrama se pregunta ¿hay tono?, lo que se está haciendo es una llamada a la función *idle de segundo nivel* encargada de la detección de tonos multifrecuencia, ver *Capítulo 6*). Lo anterior se comprueba cada vez que se entra en esta función.

  Después de comprobar lo anterior se inicia el reconocimiento de voz. Se reestima el ruido de la línea telefónica, se manda un pitido al usuario (si la variable interna ***n_usa_pitido*** está a 1) y se aplica el algoritmo *One Pass* a las muestras obtenidas por el DSP cuando nos informe que ha reconocido algo. Este proceso está dividido en estados, de tal forma que cada vez que se entra en esta función se ejecuta uno de ellos. El algoritmo *One Pass* se ejecuta en tres fases, coincidiendo cada una de ellas con un estado.

  En el caso de detección de tonos multifrecuencia se comprueba si la palabra reconocida está permitida. Si no lo está se genera el error *NO RECONOCIDO*. Asimismo, si no se detecta nada en el tiempo prefijado, se generará el error *TIMEOUT*. Ambos errores serán tratados según se indique en la sección de tratamiento de errores del programa que se esté ejecutando.

  Cuando la función acaba, bien porque se produzca el error *NO RECONOCIDO*, bien porque se ha reconocido una de las palabras permitidas, se escribe en la variable interna ***s_origen_reco*** cual ha sido el origen de la detección, 'voz' o 'DTMF' (esta información sólo será útil si están permitidos ambos modos de reconocimiento, ***s_tipo_reco = "AMBOS"***). Esto nos puede ser útil, por ejemplo, para adecuar los mensajes a reproducir en el tratamiento de error *NO RECONOCIDO*.

- **Función fin**: se libera la memoria dinámica solicitada para guardar la palabra a reconocer y su segundo candidato y se desactiva el reconocimiento, dejando al DSP en estado OCIOSO.

5.6 CAMBIO DE MODO DE RECONOCIMIENTO DURANTE LA EJECUCIÓN DE UNA APLICACIÓN

Se controla a través de la variable interna ***s_tipo_reco***, asignándole uno de los tres valores siguientes:

- **VOZ**: reconocimiento de voz.
- **DTMF**: reconocimiento de tonos multifrecuencia.
- **AMBOS**: se puede utilizar tanto reconocimiento de voz como de tonos multifrecuencia, indistintamente y ante cualquier petición al usuario. Por ejemplo, si se le está pidiendo al usuario del sistema que introduzca una cadena, dicho usuario podría utilizar la voz para decir algunos dígitos y tonos multifrecuencia para los demás.

El valor que asignemos a esta variable se puede hacer en cualquier punto del programa PRO que describe la aplicación, e incluso en varios puntos y con valores distintos.

Los valores anteriores se pueden escribir en mayúsculas o minúsculas, pero **se debe tener la precaución de**, si se utiliza la variable ***s_tipo_reco*** en la condición de una instrucción ***if***, escribirlos como se hayan definido, pues la comparación de cadenas sí distingue entre mayúsculas y minúsculas (en el apartado A.5.3.2. del *Apéndice A Manual del Usuario* se muestra una *SECCION_ERRORES* estándar, donde se utiliza el valor de la variable ***s_tipo_reco*** para reproducir un mensaje u otro al usuario en el tratamiento *TIMEOUT*).

5. FUNCIONALIDAD DE VOZ: Reproducción, grabación, síntesis y reconocimiento 5-1

5.1 INTRODUCCIóN.....5-1

5.2 REPRODUCCIóN.....5-1

5.3 GRABACIóN.....5-5

5.4 SíNTESIS.....5-7

5.5 RECONOCIMIENTO DE VOZ.....5-8

5.6 CAMBIO DE MODO DE RECONOCIMIENTO DURANTE LA EJECUCIóN DE UNA APLICACIÓN.....5-12